

DOCUMENT RESUME

ED 077 732

SE 016 290

AUTHOR Hatfield, Larry L.
 TITLE Computer-Extended Problem Solving and Enquiry, II. The Use of Computers in Mathematics Education Resource Series. Revised Edition.
 INSTITUTION ERIC Information Analysis Center for Science, Mathematics, and Environmental Education, Columbus, Ohio.
 PUB DATE Feb 73
 NOTE 39p.; Paper presented at Annual Meeting of NCTM (Anaheim, California, April 1971)
 AVAILABLE FROM Ohio State University, Center for Science and Mathematics Education, 244 Arps Hall, Columbus, Ohio 43210 (\$1.00)
 EDRS PRICE MF-\$0.65 HC-\$3.29
 DESCRIPTORS College Mathematics; Computer Assisted Instruction; *Computer Oriented Programs; Computers; Curriculum; *Instruction; *Mathematics Education; *Problem Solving; *Research Reviews (Publications); Secondary School Mathematics

ABSTRACT

The second in a series of four papers on computer use in mathematics education, this paper reviews some of the pedagogical rationales and research evidence related to the use of computers as a problem-solving tool in mathematics classes. A discussion on philosophies and objectives, effects of computer programming on mathematics achievement, motivation and attitudes, and influence on problem-solving ability are included. For other papers in this series, see SE 016 289 through SE 016 292. (Author/DT)

ED 077732

U.S. DEPARTMENT OF HEALTH
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY.

**SMEAC/SCIENCE, MATHEMATICS, AND ENVIRONMENTAL EDUCATION
INFORMATION ANALYSIS CENTER**

... an information center to organize and disseminate information and materials on science, mathematics, and environmental education to teachers, administrators, supervisors, researchers, and the public. A joint project of The Ohio State University and the Educational Resources Information Center of USOE.

FILMED FROM BEST AVAILABLE COPY

ED 077732

THE USE OF COMPUTERS IN MATHEMATICS EDUCATION
RESOURCE SERIES

II. COMPUTER-EXTENDED
PROBLEM SOLVING AND ENQUIRY

by

Larry L. Hatfield
University of Georgia

A Paper Presented at the Annual NCTM Meeting
Anaheim, California
April 1971

(Revised Edition)

ERIC Information Analysis Center for
Science, Mathematics and Environmental Education
The Ohio State University
Columbus, Ohio 43210

February 1973

Preface

Although the most publicized educational use of computers is in a tutorial CAI mode, the most popular use of computers in the mathematics classroom is probably as a problem-solving and investigative tool. This paper carefully considers the pedagogical foundations for such computer use, reviews the efforts of various project groups and research studies involving computer problem-solving usage, and summarizes what is presently known about the effects of such usage on achievement, attitudes, and problem-solving ability.

The paper also contains a critique of research and studies in the computer problem-solving area, and concludes with suggestions for further research and studies in this area.

Marilyn N. Suydam
Editor

This publication was prepared pursuant to a contract with the Office of Education, U.S. Department of Health, Education and Welfare. Contractors undertaking such projects under Government sponsorship are encouraged to express freely their judgment in professional and technical matters. Points of view or opinions do not, therefore, necessarily represent official Office of Education position or policy.

THE USE OF COMPUTERS IN MATHEMATICS EDUCATION

RESOURCE SERIES

This is a set of papers and bibliographies addressed to both mathematics teachers and mathematics educators. An introductory paper discusses the general role of the computer in education. A second paper considers the use of computers in what is at present their most widely-used role, as a tool in mathematics problem-solving. A third paper reviews research related to computer uses in mathematics education. A three-part bibliography includes selected references on the general role of computers, on language and programming, and on mathematics instructional applications.

The titles in this resource series are:

The Use of Computers in Mathematics Education:

I. COMPUTER INNOVATIONS IN EDUCATION by Andrew R. Molnar

The Use of Computers in Mathematics Education:

II. COMPUTER-EXTENDED PROBLEM SOLVING AND ENQUIRY by Larry L. Hatfield

The Use of Computers in Mathematics Education:

III. BIBLIOGRAPHY

Part 1. General Educational Role

Part 2. Languages and Programming

Part 3. Mathematics Instruction Applications

A. Teaching About Computers

B. General Uses

C. Tutorial and Practice Modes

D. Problem-Solving Mode

The Use of Computers in Mathematics Education:

IV. RESEARCH ON COMPUTERS IN MATHEMATICS EDUCATION by Thomas E. Kieren

The ERIC Information Analysis Center for Science, Mathematics and Environmental Education is pleased to make these papers and bibliography available.

Jon L. Higgins
Associate Director for
Mathematics Education

COMPUTER-EXTENDED PROBLEM SOLVING AND ENQUIRY IN MATHEMATICS

Examining the impact of technology on American education in 1969 is like examining the impact of the automobile on American life when the Model T Ford first came on the market.

President's Commission on Instructional Technology (Hechinger, 1970)

The purpose of this paper is to review some of the pedagogical rationales and recent research evidence related to the use of digital computers as instructional tools in mathematics classrooms. In particular, the approaches characterized by students writing and processing computer programs as mathematical algorithms will be included. A critique of this research and development activity will be presented. Finally, suggestions for further research efforts will be proposed.

The Approaches and Intended Outcomes

At first glance one might contemplate a very restricted set of philosophies and procedures would be connoted by the label "computer-assisted problem solving." Quite to the contrary, one finds in this literature subtle but important diversities in the philosophy, purpose, and pedagogy of having mathematics students learn to write computer programs.

Philosophies and Objectives

Foundational to probably every position is the recognition of the computer as a major force shaping the accelerated changes of our society. This leads to the contention that educated citizens of the "computer generation" should have an awareness of the capabilities and limitations of this modern tool. Bright (1965, p. 73) states "that since all professions will be radically affected by the computer, all

students will have to learn how it works and what it can do. . . by using computers as data solving tools in such subjects as mathematics, physics, and economics." In similar fashion Travers and Knaupp (1971, p. 12) argue that up through the junior high school grades "the child's contact with the computer be primarily an education in the general sense. The child would become familiar with what a computer is, what kinds of things it can do, and perhaps even more important, what it cannot do." These admonitions do not clarify how much detail about the machine our students should be expected to learn.

To be sure, there must be many levels of "knowing" a computer and what it can or cannot do, ranging from the engineer who designs its circuits to the systems programmer to the applications programmer to the citizen who mistrusts the mysterious "black box" as an error-prone, encroaching "brain." Some educators do advocate the use of selected computer science concepts along with assembler programming languages (Bolt and Wardle, 1970; Sims and Blackford, 1969; Albrecht, Lindberg, and Mara, 1969). Most mathematics educators, however, seem to subscribe to the philosophy that the activity of writing, processing, and studying the output of computer algorithms should be done to promote the development of mathematical concepts and principles, computational skills, and problem solving abilities of the student. For example, Dorn (1968) states:

It is important to keep in mind, however, that such a (computer) laboratory is best used to teach mathematical concepts and to extend the range of the mathematical topics that can be taught. The laboratory is not intended to be a device to teach programming or computer science, although some knowledge of these subjects will be a by-product of the laboratory's use (Dorn, 1968, p. 79).

Such a perspective would seek to minimize the instruction of detailed programming languages and of the machine's makeup and operation. Thus, simple-to-learn, algebraically natural programming languages would be used.

Furthermore, Dorn's viewpoint suggests that the use of computers to teach mathematics may result in some adjustments in the relative emphasis given to particular topics and objectives. Buchman (1969) argues that the increased use of calculators and computers in a wide range of society's activities is pointing to the need of further changes in the objectives of teaching mathematics. As a particular thesis for further modification of these objectives, he states;

Now, with the wide use of the electronic digital computer which can not only out-cipher the fastest human cipherer, but also can solve a wide spectrum of problems, it may be that the particular objective of developing pupils who are rapid calculators with large numbers must be replaced, finally with the objective of developing pupils who can interact with these computers (Buchman, 1969).

And in discussing questions about the ultimate objectives of mathematics education Begle (1969, p. 1) raises, as an example, the question "Should all students be expected to grasp the nature of algorithmic processes sufficiently well so that they can understand what a high-speed electronic computer can do and what it cannot do?"

Another aspect of the impact of computer use on the objectives of mathematics education is suggested in Dorn's reference to a laboratory approach (see also Dorn, 1970; Hughes, 1971, Kieren 1969a). Considerable attention is being given to investigating the pedagogy and effects of "activity" learning (Kieren, 1969b). In most of these contexts students engage in the manipulation and observation of

carefully structured concrete materials whose properties suggest important attributes of specific examples. Student behaviors often include the interpretation of given questions or problems, aimed at directing search behavior, the examination of specific instances with the concrete materials, the study of several instances to detect common properties, the formulation of a generalization about observed properties, and the testing of this generalization with additional examples. It would appear that those who view the computer as a laboratory tool see it as a device to help bridge the gap between these behaviors with concrete materials and the purely symbolic coding structures of the mathematician. The computer would be used to generate or accept those numerical instances of interest to the student, manipulate and print out the results of these manipulations, and then perhaps be used to process a more general program which incorporates the generalizing which the student has been able to accomplish. Such emphases embody the spirit of an education where one learns competence rather than particular performances. Bruner (1970) advocates such basic changes in pedagogical practice when he proposes that

education must concentrate more on the unknown and the speculative, using the known and established as a basis for extrapolation. . . . The reward for working one's way through the known is to find a new question on the other side, formulated in a new way. Let it be plain that inquiry of this kind can be made not just through 'the social sciences' but equally via the arts, literature, and philosophy, as well as by the syntactical sciences of logic and mathematical analysis (Bruner, 1970, p. 78).

Another philosophy of computer usage emphasizes the creativity of the student programmer. In this context the student would be taught how to control the machine in order to provide a context in which he

could invent algorithms which appeal to his personal interests. The Dartmouth College Project (Danver, 1969; Kurtz, 1968) is an outstanding example of promoting the computer as a creative extension of a student's intellect.

Some Pedagogical Rationales

Kieren and Hatfield (1971) attempted to identify in terms of desired student behaviors several rationales for utilizing computer algorithms in teaching school mathematics. They noted that in constructing a successful computer algorithm (program) the student must complete a careful analysis of the processes of the problem. This often requires a more intensive study of the defining attributes and restrictive conditions of the concepts involved than might otherwise occur. This careful study by the student should serve to reinforce and clarify the concepts and procedures being taught in class. In each instance the task would be to write a computer algorithm based on the student's knowledge of the concept and on a known non-computer procedure for determining instances of the concept. The central focus of this first computer use is in the designing of the algorithm; the computer output might be viewed as almost perfunctory. Of course, it does provide a means of testing the validity of the student's program and thus serves as a feedback-reinforcement mechanism.

An increased emphasis on the subsequent usage of the computer output by the student leads to the computer used as an experimental tool. In this second usage the output is used to foster inductive strategies aimed at generalizing and discovering. In these instances

a student would specify in his algorithm the generation of those cases he anticipates will be useful in identifying a pattern or consistency from which he can then state some conjecture. The computer algorithm is being characterized in this sense as a computational decision tool for generating and manipulating data to explore cause-and-effect questions as well as for identifying patterns.

These viewpoints suggest the utility of a computer program as a "dynamic" problem solving tool. Any computer program, good or bad, is an active object. With it, the student can command the computer to do something which he can observe, study, and modify. Feurzig and Papert (1968, p. 12) describe programming as a constructive problem solving process: "A solution to a problem is built according to a preconceived, but modifiable, plan, out of parts which might also be used in building other solutions to the same or other problems. A partial or incorrect solution is a useful object; it can be extended or fixed, and then incorporated into a larger structure. These remarks are true of mathematical thinking in general. But in most contexts they are too subtle to be meaningfully taught. An important example of how programming brings them down to earth is the use of the process of debugging programs as a paradigm for the crucial - but neglected - aspect of mathematical thinking that has to do with turning errors to positive advantage.

It is probably this opportunity to arrive at a satisfactory individualized program (solution) through a "successive approximation" approach that results in the high level of student motivation which many high school computer projects have reported (Haven, 1968; Johnson,

1966; Kurtz, 1968; Lund, 1969) Many students will persist through several versions in refining and extending an already successful program. Perhaps these students are exhibiting a natural, early state of what mathematicians elusively describe as "elegance."

Programming emphasizes the methods of obtaining solutions. At the same time, students have a setting where they can extend, revise, and refine their efforts to a more general algorithm aimed at processing entire classes of problems. The computer will serve as a precise judge of each individual approach, whatever the level of sophistication.

The student learns to consider extreme or special cases where the procedure he is designing fails. Since a program requires a logical organization, a student encounters various patterns of reasoning in the decision structure of the program and in the computer output. Incorrect output might be used to trace at what values the procedures of the program began to show errors. Correct sequencing or arrangement of conditional branching involves the logical connectives and and or used to find solution sets for conjunctive and disjunctive systems of mathematical conditions.

The elusive notion of variable becomes more operant as the student deals with a system in which a symbol (variable) denotes any member of a set of replacements but which at any given instant during the processing of the program must actually be assuming one of the permissible values. As the student constructs or tests his algorithm, he can interpret variables in terms of currently assigned values. Thus, generalized procedures can become easier to design or comprehend.

Numerous excellent accounts of the pedagogical advantages of using

a computer as a tool to promote student concept reinforcement and problem solving in school mathematics are available in recent issues of The Mathematics Teacher. One classroom teacher's account is particularly descriptive. Hughes (1971) notes the need . . . lly selecting concepts and problems to which computer use naturally makes a mathematical contribution. Using a study of Gauss's Theorem for constructible regular n -gons as a problem setting, she reports:

In no case was a first program run successfully, although nearly all were at least partially successful. As students analyzed their programs to make corrections, they saw they had not completely understood Gauss's concept and were obliged to reexamine his constraints to see which ones they had not built into their algorithms. The writing and rewriting of these programs furnished a clear-cut example of the power of computer assistance to clarify a concept by requiring a student to verbalize it accurately and completely. . . the students who attempted it found it difficult but not impossible, the kind of problem that seizes hold of the mind so that the only way to be free of the problem is to solve it. All students in the twelfth grade and a great many in the eleventh grade found solution, although these differed greatly in generality. Students were severe critics of each other's programs, and many wrote additional programs with new data lists to prove that their algorithms were completely general. They were watchful for cases where the algorithm produced the right number of sides for the wrong reason and were critical of programs that required excessive computer time to run (Hughes, 1971, pp. 156-7).

Considerable activity with computer programming at the college level can be found. The Committee on Undergraduate Programs in Mathematics (CUPM) of the Mathematical Association of America (1964, 1969) has offered recommendations supporting work in computing. A recent CUPM Newsletter (1969, p. 2) cited the following reasons for using the computer in teaching calculus:

1. Freshman calculus, because of the nature of the subject and its placement in the undergraduate curriculum is a good

course for the introduction of computers into the student's thinking and working habits.

2. Overall student interest and comprehension is increased as learning is turned into a more active, less passive experience, and more emphasis is given to the "constructive" and algorithmic aspects of the calculus. The computer gives the student opportunity for experimentation.

3. Problems become more real, more challenging, more interesting, and less tedious when programmed for a computer.

The Center for Research in College Instruction of Science and Mathematics (CRICISAM) has developed a computer-oriented calculus text which, according to a recent CRICISAM Newsletter (1970), is being used this year in approximately seventy universities and colleges. No indication of plans for controlled experimentation have been noted. Finally, Dorn and Hoffman of the University of Denver Research Institute are directing a Computing and Mathematics Curriculum Project which has prepared instructional materials for a computer extended calculus treatment.¹

Thus, there appears to be reasonable diversity in the purposes for teaching students how to program a computer. While there may be continued disagreement about how much emphasis should be placed on computer science topics or which programming language should be taught or when students should be taught how to program, it would seem that engaging mathematics students in using the computer has fired the imagination and enthusiasm of many mathematics educators. But what research evidence can be found to support the claims offered by computer enthusiasts? The experimental studies and project reports are reviewed in the following section.

¹For further information, write the Department of Mathematics, University of Denver, University Park, Colorado 80210.

Project Efforts and Research Results

A variety of exploratory projects has emerged in recent years devoted to the study of the feasibility and advisability of using computers and computer programming in mathematics instruction. No effort will be made here to exhaustively review all of these projects. Instead those known developments which offer or intend to offer either "conclusion-oriented" or "decision-oriented" findings (Cronbach and Suppes, 1969) will be included at this time.

The Effects of Computer Programming on Achievement in Mathematics

Most investigators have sought to examine the following question: Does the activity of writing computer programs and the study of computer output affect the achievement of mathematics students when the problems to be programmed are directly related to the regular curriculum?

Danver (1969) reports on the Dartmouth College secondary schools project. Eighteen New England secondary schools have been participating in a three-year NSF project to experiment in computer training and use. The main purpose cited is to demonstrate the use of the computer as a broad aid to secondary education without requiring major curriculum changes or extensive teacher retraining. Although no controlled studies of student achievement were conducted, Danver (1969, p. 15) states, "Many teachers claim that the students who become interested in computer applications to their coursework learn far more than the other students because of it. They use it to explore and to answer questions unanswered in class." Additional comments are reported from project teachers who all support computer access for students.

Project LOCAL has involved five Massachusetts School systems in a two-year project with objectives that include experimenting with students using the computer as a problem-solving vehicle. Haven (1970) reported in a preliminary release the results of two independent statistical analyses involving one junior and one senior high school sample. Univariate and multivariate analysis of variance of measures of mathematics achievement, attitudes toward computer assisted methods of teaching, and general changes in motivation were completed. The Cooperative Mathematics Achievement Tests for Algebra, Trigonometry, and Analytic Geometry were administered pre- and post-treatment to the high school sample. Reporting on the univariate analysis, Haven (1970) states that

the mathematics achievement subareas were the portion of the evaluation wherein the most probable effects of the treatments were expected. Impressive differences were indeed found, in treatment effect and in pre-post measurement, all significant at the .05 level. It appears clear that the effects of computer-assisted instruction have a direct influence upon the measures of achievement in the areas of mathematics (Haven, 1970, p. 5).

Multivariate analyses appeared to substantiate some notion of treatment effects with only the Analytic Geometry results being significantly different. Haven (1970, p. 5) summarizes with the observation that "although the cause of such high significance in the area of analytical geometry (with no significance in any other math subareas) is unknown, the treatment effects seem positive." Only small differences were found in the analyses of the STEP Mathematics Test data from the junior high school samples.

The computer was used as a problem-solving tool in two studies conducted as a part of the University of Minnesota CAMP Project

(Johnson, 1966). Kieren and Hatfield (1971) report on the results of replicated experiments involving grade seven and eleven students. Comparisons involved several performance measures obtained from constructed and standardized tests. In the first experiment in grade seven, significant treatment effects favoring the non-computer group were observed on the initial unit test (numeration systems) with the greatest difference occurring at the low previous-achievement level. The learning of this mathematical material seems to have been confounded by the concurrent introduction of the BASIC programming procedures. A revision of the approach taken in this unit resulted in no differences between the treatments of the second year. In the grade seven experiment of the second year, significant treatment effects favoring the computer group were observed on one post-unit test (elementary number theory) and on two post-treatment tests (Contemporary Mathematics Test, Junior High Level, and a problem-solving test). The treatment effects obtained on the test following the elementary number theory unit were viewed as especially important.

This unit was recognized as being a particularly significant setting for use of the computer. The calculating power and speed of the machine provided the incentive to construct algorithms to handle nearly every concept and process studied. Thus, the activity of algorithm design probably functioned to reinforce and clarify ideas. Furthermore, this mathematical material especially lends itself to promoting the experimental power of the tool. Therefore, students were challenged to research several topics with the computer. This approach seemed to enhance the productivity of students at all ability levels. This observation is supported by the results on the unit test (Kieren and Hatfield, 1971, p. 11).

In the grade eleven experiments significant differences favoring the computer treatment occurred on one post-unit test (quadratic functions) and on one post-treatment measure (Contemporary

Mathematics Test, Advanced Level). On the trigonometry unit test, however, the regular treatment group achievement significantly surpassed that of the computer treatment. No significant treatment by previous-achievement level interactions were observed for all measures at either grade level. However, the results suggest that the average and above-average seventh grade achievers seemed to benefit relatively more from the computer treatment while the grade eleven study suggested a positive differential effect for its average achievers. A null hypothesis of no difference in the proportion of students correctly responding to a test item was tested for each of the tests. In the second year the grade seven computer group scored significantly better on 25 items while the non-computer students were favored on 13 items. This hypothesis was rejected for 43 items in the grade eleven replication with the computer group favored in 16 items and the non-computer group in 27 items.

Wallace (1968) studied the impact of computer mathematics on the learning of high school trigonometry. He involved classes of eleventh and twelfth grade students in three separate treatments: Class I was taught trigonometry conventionally; class II was taught a semester course in computer mathematics, then trigonometry as for Class I; and Class III was taught fifteen weeks of trigonometry, then was given review for three weeks during which flow chart and elementary computer techniques were used as yet another means of learning trigonometric relations and problem solving. The Class III students showed significantly more gain in knowledge of trigonometry. No explanation is offered for the observed result. Wallace (1968, p. 3540-A) concludes

that "the use of flow charts and algorithmic methods in teaching mathematics appears to fortify conventional teaching methods, with the result that higher learning rates are attained." It should be noted that this experimental result is somewhat in dissonance with Kieren's (1968) results in the grade eleven trigonometry unit of his experiments.

The widespread efforts to study computer programming in school mathematics are not restricted to the United States. Lund (1969) describes the results of a pilot study conducted in The Hague, Netherlands, with seventh and eighth grade mathematics students. After initial instruction in writing mathematics programs in the BASIC language, these subjects were randomly assigned to three alternating experimental and control groups at each grade level. During a particular instructional unit control classes did not utilize computer methods to study the same content as the computer treatment groups. The performances of the computer treatments were higher, but not significantly so, on all tests given subsequent to each unit. Additional research is planned when further curriculum usages are identified and inservice training of faculty has been completed. Lund (1969, p. 4) concludes: "In summary, our experience with a time-sharing system here in Holland has convinced us that computers offer a valuable contribution to the learning and teaching of mathematics." It may be appropriate to note that further European activity is also expected as a result of the international Project for Computing in the Schools.²

²For information contact Mr. Bryan Thwaites, 11 Thistle Street, Edinburgh EH2 1DG, England.

At least one study has attempted to examine a programming approach across mathematics classes at the seventh, eighth, twelfth, and college freshman levels. Washburn (1969) coordinated programming exercises with the mathematical topics of a particular course in an effort to study the effect of requiring the student to re-examine his knowledge and understanding of one or more concepts required by his course. The students in the experimental classes were taught the elements of the CUPL programming language as they were needed for the programming exercises. The students in both computer and control groups were pre-and post-tested on all topics. As a result of his experiment, Washburn (1969, p. 5179-A) concludes: "The writing, execution and correction of computer programs can strengthen one's understanding of mathematical concepts. This gain is independent of one's age and level of mathematics achievement. Although students of higher intelligence tend to derive greater benefit, students of average and lower intelligence benefit as well." It was not reported whether this ability difference is consistent across all grade levels.

Several controlled investigations have been completed with samples of college students. While most studies at this level have focussed on the introductory course in calculus, Morgan (1968) sought to examine the role of the digital computer in a college general education course in mathematics. He attempted to identify topics from computer science with inherent mathematics appropriate for general education students. The direct objectives were to provide and maintain basic mathematical skills, to develop a familiarity with and an appreciation of elementary aspects of computer science,

and to establish positive attitudes toward mathematics. It was not clear to this writer exactly what was the nature of the treatment; in particular, it was not indicated whether subjects were simply told about selected computer applications or whether they were taught a programming language and then used it to write or interpret programs stemming from applications. In any event, Morgan (1968, p. 72-A) reports that "results of the standardized post-test were significantly better than the pre-test. It appears that general education students can enhance their mathematical competence when the content is integrated with computer-based application."

Fielder (1969) compared course achievements for two groups of Analytic Geometry and Calculus I students. Four mathematical concepts were selected for the investigation: functions, limits and differentiation, iteration, and integration. One group wrote FORTRAN IV programs to solve homework problems while the other class completed homework in the usual way. Four constructed tests were administered as pre- and post-test for each concept phase of the experiment. Applying analysis of covariance at the 0.10 level of significance, no statistically significant difference in achievement between the treatment groups was found. Of course, Fielder (1969, p. 3911-A) concludes that "students learn mathematical concepts just as well by computer programming as they do by solving problems in the usual homework structure."

Holoien (1970) points out that no computer terminal was used in the classroom in Fielder's study. Viewing such contact as important, he arranged to make an instructor's demonstration terminal available to his experimental calculus groups. These students learned

a format-free version of FORTRAN to complete the programs required for about half of the homework exercises. Students submitted hand-written programs to be keypunched and processed in the campus computer center with about a one-day turn-around; that is, students generally did not use the terminal in the classroom for "hands on" interactive executions of their programs. Holoien observes that unavailability of a nearby time-sharing computer system with a conversational programming language such as BASIC precluded implementing a more desirable terminal facility. Although the effect due to instructor is confounded in the experimental design, he obtains some interesting results.

The hypothesis of no difference in scores on achievement tests for two treatments was tested for each of three unit tests, the final examination and for all four treated as a single test. A near-significant F-value ($.05 < p < .10$) was observed on the test for the first unit, favoring the computer treatment. He was able to reject the null hypothesis ($p < .025$) in favor of the computer group for the second unit test which dealt with evaluating functions and the limit of a function at a point. Holoien (1970, p. 59) observes: "It is worth noting that some of the concepts that especially lend themselves to computer programming appear in the part of the course sampled by Unit Test II." A significant difference favoring the computer group was also observed on the pooled-data analysis. He also noted interaction effects on two unit exams and on the pooled exams favoring the computer treatment for students of lower mathematical ability. Holoien (1970, p. 65-6) notes that "such results are probably what one would expect; that is, higher-ability students learn mathematics

concepts regardless of teaching methods used whereas those of lower ability seem to do better when special learning aids are used."

The two other known studies of a computer-oriented approach to calculus yielded results which support the significance Holoien observed. Bell (1970) used the last six weeks of an introductory calculus course to conduct his experiment. Two sections of students studied calculus using a prepared calculus manual. These materials were identical for the two groups except that the Experimental Materials contained six computer-oriented problem sets to be solved by writing and executing computer programs. In place of these problem sets the Control Materials contained six problem sets composed of non-calculus problems to be solved by writing and executing computer programs, and calculus problems to be solved without using a computer. Thus, both groups in this study were taught to use the computer. Hypotheses of no differences in performance involved examinations of techniques of calculus and understanding of calculus concepts administered immediately after instruction and one month after instruction. Using achievement data from four examinations, these hypotheses were tested in three ways: (1) using t-tests for differences between means of scores; (2) using analysis of covariance for differences between means of scores by controlling for differences in SAT-Mathematics scores between the groups; and (3) after selecting matched subgroups of 36 students from each of the groups, performing t-tests for differences between means of scores. Hypotheses dealing with knowledge and retention of techniques of calculus were not rejected for any of the four examinations. However, significantly

different performances in immediate and retained understanding of calculus concepts were observed for all four examinations in favor of the experimental group. Bell (1970, p. 1096-A) summarizes: "The conclusions of the experiment support the hypothesis that a computer-oriented approach to calculus is an effective method to promote understanding of concepts and to increase students' interest in calculus, and does not interfere with students' learning to apply techniques of calculus."

The four concepts of function, limit, derivative and application of derivative (Newton's method of approximating solutions of an equation) were featured in the computer homework assignments in an investigation completed by Bitter (1970). Two introductory calculus classes taught by the same instructor were compared at each of three participating colleges. Students in the computer-extended groups solved calculus homework assignments by writing BASIC programs which they executed via time-sharing remote terminals. Using the Cooperative Calculus Mathematics Test, Forms A and B, and a pre-test/post-test control group design, Bitter tested an achievement null hypothesis for Part I (differential) and Part II (integral) scores. Part I items dealt with the computer-extended topics while Part II items did not include any questions related to the computer applications. Bitter reports that the subjects who were provided with the computer-extended instruction scored significantly higher on Part I while no significant differences were found on the Part II data. An interesting sidelight to this study is that, disregarding the treatment effect, the female students achieved higher than their male

counterparts. No differences in treatment effect for the sexes was noted, however.

It should certainly be apparent, with its short history and additional resource requirements, that investigations with computer-extended problem solving are reasonably scarce. The studies reviewed here offer some initial evidence across mathematics instruction from junior high school through beginning calculus. The conclusions which one might attempt to state from the achievement findings can at best be vague and tentative. Does computer-extended instruction affect the achievements of students with respect to the regular objectives and content of mathematics instruction? Overall, there appears to be some support for the claim of improved concept and principle knowledge and use.

It would also seem that students of junior high school age and older can and will learn to write computer programs to study mathematics, although some concerns should be shown for how the programming is introduced and which language is chosen. All of the studies reported here were administered within the framework of the experimental treatment instruction in the elements of a programming language and the techniques of constructing a program. Several different languages were used as were several different degrees of student access to the computer. Although not a definitive result, it would appear that treatments which used a conversational programming language and which provided direct student access through a computer terminal were more often able to show significant treatment results favoring the computer groups. Hunka (1970) observed that the language

APL was quite easy for elementary and junior high students to learn, but teachers at the senior high level reported student difficulties in learning APL. This surprising report may be explained in part by examining the type of usage made of the computer at each level. Teachers and students at the lower levels primarily wrote functions for drill and practice work while the stress on algorithmic solutions to problems in the high school led Hunka to state the "even though some (high school) teachers had taken the basic computing science course at the university, which included APL, knowing APL operators and being able to construct educationally useful functions in support of the curricula is not always easy (1970, p. 8)."

Motivation and Attitudes Toward Computer-extended Problem Solving

One of the outcomes lauded by educators who are computer proponents is the increase in interest and enthusiasm of students toward doing mathematics in a computer algorithm design context. Nearly every development project report will contain numerous descriptions of students who become "turned on" by the computer. For example, Haven concluded an anecdotal account with the following:

The force of the computer's motivating power is often so strong that it leaves teachers a little flabbergasted. In fact, teachers are finding that with some students they must guard against "over-involvement". Michael Wolff, in a recent article in Science and Technology magazine, coined a term which very aptly sums up this situation. It seems that two institutions spawned by modern-day affluence, the ski-bum and the surfing-bum, are now being joined by another rapidly growing group, the "computer-bums"; that is, students who would rather work with the computer than do almost anything else. However, if handled correctly this effect is all to the good, as it provides the school a valuable tool to promote learning by making subject matter interesting and meaningful (Haven, 1968, p. 44).

Project LOCAL subsequently attempted to evaluate subject's attitudes toward computer assisted methods of teaching. The analysis performed at the high school level included the use of the Kuder Preference Record, Vocational Form C, with data gathered before and after the year's treatment. Significant treatment effects due to students shifting their career choices demonstrably were shown in the univariate analysis. However these significant values did not hold up under the more sensitive multivariate analysis (Haven, 1970).

Several of the studies described in the previous section included efforts to assess student attitudes. Washburn (1969) used a testing instrument at the end of his experiment to measure student attitude toward the computer enriched approach. "Not only is there a strong positive student attitude toward the Computer Enriched Mathematics Program approach, there is an indication that this approach can also improve student attitudes toward mathematics in general. These attitudinal trends are independent of one's age, level of mathematics achievement and intelligence (Washburn, 1969, p. 5179-A)."

Holoien (1970) used the Rabinowitz Attitude Inventory as a pre- and post-test in his calculus classes. He was unable to reject the hypothesis of no change in attitude toward mathematics over the project period. Written comments were solicited from the experimental classes at the end of the experiment. Holoien (1970) summarizes his descriptive treatment of these student comments:

It is quite apparent that a large majority of student to whom computer-supplemented calculus was taught were favorable to the use of the computer. The two criticisms mentioned most frequently by students who were favorable toward the use of a computer were 1) the programming part of the course

was too concentrated and 2) a computer terminal should have been available for students to use outside of class (Hickien, 1970, p. 71).

Discussing these observations, the investigator conjectures that if students had had more opportunity to use a terminal themselves, their attitudes might have changed enough to be detected by an attitude scale.

The only other study which reported any serious attempt to examine attitudes of students toward computer-oriented approaches was the calculus experiment conducted by Bell (1970). From the results of a student questionnaire used to solicit opinions concerning the experiment, he concludes that the computer treatment increased students' interest in the calculus. No further details are provided as a basis for this conclusion.

The interpretation of these results must remain even more tenuous than those offered about achievement effects. Clearly, mathematics educators who utilize computers as problem-solving tools consistently applaud the motivating power of this instructional approach. Yet the experimental evidence awaits generation.

Influence on Problem Solving Ability

Because the activity of writing, processing, and correcting computer programs is often characterized as problem solving behavior, it would seem that the study of problem solving outcomes would be of explicit interest to researchers in this area. Although particular tasks included in the achievement tests used in the studies reported here may indeed involve problem solving behaviors, almost none of these investigators chose to explicitly examine a problem solving construct.

The Differential Aptitude Test, Abstract Reasoning, served as one of the measuring instruments in the Project LOCAL evaluation (Haven, 1970). Used on a pre- and post-test basis for both the junior and senior high school samples, the difference scores were subjected to both univariate and multivariate analyses. Only marginal indication of a significant treatment effect was noted at both school levels following the univariate analysis. Multivariate analysis, using an IQ measure as an independent variable, did not support this tentative significance. A third analysis of the DAT scores was performed to compare differences between the two school levels. Of course, large differences would be expected due to the level of education differences between the two schools. Haven (1970) reports significant results from treatment effects, pre-post, and between levels of education and indicates that cursory analysis points toward a more significant effect at the high school level for the influence of the computer-extended instruction upon the abstract reasoning ability.

Hatfield (1969) developed a construct definition of problem solving behavior which was used by a panel of mathematics educators in identifying which test items were "problems". All items were subjected to a test of no differences in the proportions of students responding correctly to an item. Of the 38 items where performances differed significantly between the two treatments of the second year, 16 had been classified as "problem" items. On 12 of these items the computer treatment group was favored while the non-computer grade seven students were favored on 4 "problem" items. One post-treatment test was included to examine student abilities to handle unfamiliar problems. The

Thought Problems Test presented detailed word problems requiring careful analysis, reasoning, and insightful solutions. These items did not involve direct application of the concepts or processes of this grade seven curriculum nor any reference to computers or computer programming. The significant difference favoring the computer group supports the hypothesized improvement of generalized problem solving abilities. However, the modest results of these two studies (Haven, 1970; Hatfield, 1969) fail to provide even minimal certainty of the potential of this computer use on problem solving abilities.

Critique of Previous Studies and Suggestions for Further Research

This review has attempted to identify and report the findings of the research efforts directed toward the systematic study of computer-extended mathematics instruction. About 15 known projects were contacted and nine doctoral dissertations and five project reports were located for the information reported in this review. This certainly represents a minimal content and temporal base from which to extrapolate research results and direction. However, the inconsistency between the near-feverish contentions of positive learning effects of the ardent computer supporters and the lack of definitive, detailed knowledge offered by the current literature prompts an effort at this point to critique and suggest.

Criticisms

Shulman (1970) lucidly argued that educational researchers must undertake a dramatic reconstruction of their most basic tactics of investigation if the goal of an empirically-based discipline of

education is ever to become a reality. Mathematics education researchers, including the authors of the research studies reported above, will find the perspectives offered by Shulman particularly appropriate. To use one of Shulman's incisive figuratives (1970, p 392), the research reviewed in this paper appears to represent another of the "ever-so-precise figure eights, carefully retracing well-worn patterns" which is so characteristic of educational research. That is to say, despite the visionary use of the computer as an instructional innovation, these researchers resorted primarily to classical, over-simplified, under-defined, under-controlled, non-theory-based approaches.

The blunt question "Is curriculum A better than curriculum B?" should have been finally abandoned in the sixties after the flurry of dissatisfying comparisons of the "new" and the "old" mathematics programs. Yet the conclusions which these computer-oriented investigations are able to provide to either the theory-builder or the classroom practitioner lend little clarity or new knowledge to the complex task of determining the "what, when, to whom, how, and how much" of classroom instruction and learning. This is not to say that no new information has resulted from these investigations. Certainly there are general indicators that will at least support teachers and researchers who would undertake their own experimentation in the use of computers and computer programming in the mathematics classroom. These studies have at least fulfilled a feasibility claim; that is, a relatively small (and perhaps select) group of teachers were able to involve students in learning mathematical content while engaging in writing, processing, and using computer algorithms and the computer output. However, the

predictability and generalizability of the instructional methods and learning results of these studies are simply unknown. Tukey (1969, p. 85) observed that the use of statistical testing was never meant to serve as a substitute for replication; "Repetition is the basis for judging variability and significance and confidence. Repetition of results, each significant, is the basis, according to R. A. Fisher, of scientific truth." Of course, the results of the four earlier-sited calculus investigations (Bell, 1970; Bitter, 1970; Fielder, 1969; Holoien, 1970) taken collectively do represent a global sort of replication. But certainly the variety of methods, materials, learning tasks, durations, testing instruments, experimental designs, statistical techniques, and findings precludes any generalized adoption of computer-extended calculus as a direct experimental result.

These studies might be characterized as "formulative" evaluations of prototypes of computer-extended mathematical instruction. Certainly it can be argued that such efforts are appropriate to dissertation-level research. At the same time, the theoretical and methodological shortcomings of the investigations are the focus of this criticism. Treatments are often only vaguely defined. Although the learning of a programming language was central, no investigator gave any measured evidence of the extent of student mastery of this knowledge and skill. And even though the writing of programs for mathematical concepts and problems was the major instructional variation, no evidence is given for how many correct programs overall were written, how many trials to a correct program for each problem assigned, or the qualitative nature of the errors or successful programs for particular types of achievers.

Sample sizes are often very small and information regarding random assignments to treatments is often lacking. The instructional time was sometimes not the same for computer and non-computer treatments. On the other hand, Bloom (1968) presented strong arguments for educators to control achievement level while varying instructional time and resources in order to permit and to promote "mastery learning". Of course, none of these "formative" studies discussed here employed the strategies of this model (see Shepler, 1970). In one study the control group actually was asked to solve non-calculus problems with computer programs while also solving calculus problems without the computer. The nature of these non-calculus problems is unknown. One can only speculate on the interference effects provided by these additional, apparently non-essential tasks. Only one study explicitly sought to measure retention of knowledge although others used what appeared to be comprehensive final examinations as post-treatment measures. The duration of the experiments did seem adequate to have induced measurable change in the learners. Statistical techniques generally were viewed as appropriate contingent upon the satisfaction of the underlying assumptions. In particular the use of analysis of covariance is questionable unless an investigator can satisfy the assumptions of random assignment of subjects to treatments, independence of covariate and treatments, and no treatment-slope interaction. These contingencies were not always made clear.

Reasonable uncertainties exist about the validity of some of the measuring instruments. Even in those instances where the investigators might have designed tests to reflect the intended objectives of

instruction, performance statements describing terminal student behaviors are absent. Often standardized tests of questionable appropriateness were employed without theoretical justification for such selections. And, as has already been noted, almost none of these investigators have replicated results to report.

In summary these investigations generally lack the theoretical rationale, the careful specification of treatments, the detailed monitoring of treatment effects on those aspects crucial to the experiment, and the objectives-valid measurement of terminal and retained performances which should characterize "formative" research. At the same time, these few initial studies provide tentative information to teachers and to researchers as they consider computer-extended mathematical instruction.

Suggestions for Further Research

The criticisms submitted above invite the consideration of suggested alternatives for making progress in developing a predictive knowledge base for using classroom computers. Ideally, teachers need to know, to some specified degree of confidence, when to teach and use programming, for which types of students, for which content selections, and how this should be done (which programming language, what kind of computer access, which combinations of classroom pedagogies, etc.). And throughout all such specifications, the intended learning outcomes should be clearly explicated with performance terms and in a taxonomic specification (see Wilson, 1971). Furthermore, studies of mathematical learning in any context should employ existing theories of learning and child development in their planning, and then eventually feed back

empirical results which support or lead to modifications of these theories.

The task of supplying such information to teachers and theories appears justifiably complex. Shulman (1970) characterizes most schooling situations as highly complex and variegated activity, involving students, subject-matter and sources of instruction.

Any research which purports to deal systematically with phenomena at this level of complexity must itself reflect an appropriate level of complexity. The ideal research setting. . . must be (1) experimental; (2) longitudinal; (3) multivariate at the level of both independent and dependent variables, and consistent with that; (4) differential, in that the interactions of the experimental programs with the students' entering individual differences are treated not as error variance, but as data of major interest in the research (Shulman, 1970, p. 387).

What, then, is required are considerable more detailed efforts if we are to move beyond studies where the primary conclusion is that "more research needs to be done". Of course, a more extensive program of research will also require small, one-shot, "clinical-level" studies whose function will be to generate hypotheses. Such work may, in fact, become the coordinated contributions that doctoral dissertations may make. But we must realize that the programmatic research suggested here will require that we no longer accept the individual doctoral thesis as the prototype for scholarly efforts in the field. We will need to develop a functioning cooperative effort, perhaps even at the inter-university level, with dissertations planned as part of a broader, coordinated general program of research engaged in collectively by their thesis supervisors. Models have been proposed for such coordinated team efforts for programs of research (see Romberg and Devault, 1967; Aiken, 1970).

To this date, a handful of mathematics educators have shown research interests in the learning of mathematics with the computer serving as a problem solving tool. This paper's purpose was to summarize and discuss their work. Reflecting upon our clumsy gropings and modest successes, can we now consider new cooperative paradigms?

REFERENCES CITED

- Aiken, Lewis R. Jr. "Attitudes Toward Mathematics," Review of Educational Research, Volume 40, October 1970, pp. 551-596.
- Albrecht, Robert L., Eric Lindberg, and Walter Mara. Computer Methods in Mathematics, Reading, Massachusetts: Addison-Wesley Publishing Company, 1969.
- Begle, E. G. "The Role of Research in the Improvement of Mathematics Education," an unpublished paper presented to the First International Congress of Mathematical Education, Lyon, France, August 1969.
- Bell, Frederick H. "A Study of the Effectiveness of a Computer-oriented Approach to Calculus," an unpublished doctoral dissertation, Cornell University, 1970 (Dissertation Abstracts, Volume 31'3, p. 1096-A).
- Bloom, Benjamin S. "Learning for Mastery," Evaluation Comment, Volume 1, May 1968.
- Bolt, A. B. and M. E. Wardle. Communicating with a Computer, London: Cambridge University Press, 1970.
- Bright, R. Louis. "Enormous Role Seen for Computer," Washington Monitor, issue unknown.
- Bitter, Gary G. "Effect of Computer Applications on Achievement in a College Introductory Calculus Course," an unpublished doctoral dissertation, University of Denver, 1970 (Dissertation Abstracts, Volume 31'10, p. 109-B).
- Bruner, Jerone. "The Skill of Relevance or the Relevance of Skills," Saturday Review, April 18, 1970, p. 66ff.
- Buchman, Aaron L. "The Use of Calculators and Computers in Mathematics Instruction in New York High Schools," School Science and Mathematics, Volume 19, May 1969, pp. 385-392.
- Center for Research in College Instruction in Mathematics. "Calculus, A Computer Oriented Presentation," CRICISAM Newsletter, No. 1, November 1970.
- Committee on Undergraduate Programs in Mathematics. Recommendations on the Undergraduate Mathematics Program for Work in Computing. Buffalo, New York: The Mathematical Association of America, 1964.

- Committee on the Undergraduate Program in Mathematics. Calculus with Computers. Berkeley, California: Mathematical Association of America, August, 1969.
- Cronbach, L. J. and P. Suppes (Eds.) Research for Tomorrows Schools: Disciplined Inquiry for Education. London: Macmillan, 1969.
- Danver, Jean H. "Interim Report II," an unpublished report of the Dartmouth College "Demonstration and Experimentation in Computer Training and Use in Secondary Schools" project, 1969.
- Dorn, William S. "The Computer: A Mathematics Laboratory," Journal of Educational Data Processing, Volume V, Spring 1968, pp. 67-79.
- Dorn, William S. "Computer-extended Instruction: An Example," The Mathematics Teacher, Volume LXIII, February 1970, pp. 147-158.
- Feurzig, Wallace and Seymour Papert. "Programming Languages as a Conceptual Framework for Teaching Mathematics," Proceedings of the NATO Conference on Computers and Learning, Nice, France, May 1968.
- Fielder, Leigh A. "A Comparison of Achievement Resulting from Learning Mathematical Concepts by Computer Programming Versus Class Assignment Approach," an unpublished doctoral dissertation, The University of Oklahoma, 1969 (Dissertation Abstracts, Volume 29'11, p. 3910-A).
- Hatfield, Larry L. "Computer-Assisted Mathematics: An Investigation of the Effectiveness of the Computer Used as a Tool to Learn Mathematics," an unpublished doctoral dissertation, University of Minnesota, 1969 (Dissertation Abstracts, Volume 30'10, p. 4329-A).
- Haven, Robert N. "Project LOCAL - First Year," an unpublished project report, 1968.
- Haven, Robert N. "Project LOCAL - 1970 Evaluation," an unpublished project report, 1970.
- Hechinger, Fred M. "Time to Teach Those Teaching Machines," The New York Times (Education Section), Sunday, February 15, 1970.
- Holoien, Martin O. "Calculus and Computing: A Comparative Study of the Effectiveness of Computer Programming as an Aid in Learning Selected Concepts in First-year Calculus," an unpublished doctoral dissertation, University of Minnesota, 1970 (Dissertation Abstracts, Volume 31'9, p. 4490-A).
- Hughes, Helen S. "Gauss, Computer-Assisted," The Mathematics Teacher, Volume LXIV, February 1971, pp. 155-166.
- Hunka, S. "Use of APL Computer Terminals in the Edmonton Public Schools, an interim report, April 1, 1970 - June 30, 1970," an unpublished project report available from Division of Educational Research, University of Alberta.

- Johnson, David C. "Preliminary Reports, No. 1 and 2," unpublished reports of the Computer-Assisted Mathematics Project, 1966.
- Kieren, Thomas E. "The Computer as a Teaching Aid for Eleventh Grade Mathematics," an unpublished doctoral dissertation, University of Minnesota, 1968 (Dissertation Abstracts, Volume 29'10, pp. 3526-A).
- Kieren, Thomas E. "Quadratic Equations - Computer Style," The Mathematics Teacher, Volume LXII, April 1969a, pp. 305-309
- Kieren, Thomas E. "Activity Learning," Review of Educational Research, Volume 39, October 1969b, pp. 509-522.
- Kieren, Thomas E. and Larry L. Hatfield. "Computer-Assisted Problem Solving in School Mathematics," an unpublished paper prepared for the symposium, Computers in Mathematics Education, at the AERA Annual Meeting, February 1971.
- Kurtz, Thomas "Interim Report," an unpublished report of the Dartmouth College "Demonstration and Experimentation in Computer Training and Use in Secondary Schools" project, 1968.
- LaFrenz, Dale E. and Thomas E. Kieren. "Computers for All Students: A New Philosophy of Computer Use," School Science and Mathematics, Volume LIX, January 1969, pp. 39-41.
- Lund, Charles H. "An Encounter with a Computer - International Style," an unpublished project report for The American School of the International Schools of the Hague, 1969.
- Morgan, Richard T. "The Role of the Digital Computer in a General Education Course in Mathematics," an unpublished doctoral dissertation, Columbia University, 1968 (Dissertation Abstracts, Volume 30'1, p. 71-A).
- Romberg, Thomas A. and M. Vere DeVault. "Mathematics Curriculum: Needed Research," Journal of Research and Development in Education, Volume 1, 1967, pp. 95-112.
- Shepler, Jack. "Parts of a Systems Approach to the Development of a Unit in Probability and Statistics for the Elementary School," Journal for Research in Mathematics Education, Volume 1, November 1970, pp. 197-205.
- Shulman, Lee S. "Reconstruction of Educational Research," Review of Educational Research, Volume 40, June 1970, pp. 371-396.
- Sims, G. F. and W. A. R. Blackford. "Computing in Schools: The Warley Experiment," The Times Educational Supplement, Volume 2813, April 18, 1969, p. 1262.

- Travers, Kenneth J. and Jonathan E. Knaupp. "The Computer Revolution Needs YOU!" The Arithmetic Teacher, Volume 18, January 1971, pp. 11-17.
- Tukey, J. W. "Analyzing Data: Sanctification or Detective Work?" American Psychologist, Volume 24, 1969, pp. 83-91.
- Wallace, David C. "The Impact of Computer Mathematics on the Learning of High School Trigonometry and Physics," an unpublished doctoral dissertation, University of Texas at Austin, 1968 (Dissertation Abstracts, Volume 29'10, p. 3520-A).
- Washburn, Robert M. "CEMP - A Computer Enriched Mathematics Program," an unpublished doctoral dissertation, Cornell University, 1969 (Dissertation Abstracts, Volume 30'12, p. 5179-A).
- Wilson, James W. "Evaluation of Learning in Secondary School Mathematics," Handbook of Formative and Summative Evaluation of Student Learning (Bloom, Hastings, Madaus). New York: McGraw-Hill Book Company, 1971, pp. 743-696.